EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	1	US20050155024A1	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/25 16:16
S2	93	("20050155024" "20050193373" "6167453" "6470494" "6585779" "6732108" "20030033310" "6535894" "20020178439" "20030018614" "5966702" "6536035" "6530080" "6584612" "6684387" "6802054" "6918106" "20020042833" "20020073063" "20020099865" "20020147735" "20030009743" "20030061247" "20040221268" "20040088681" "20040123285" "5937411" "6324637" "6862683" "20050049998" "6230184" "6480880" "20010037356" "20060020932" "6980979" "20030236657" "20050097082" "20060136401" "6289512" "6542887" "6633892" "6718364" "6757685" "7107592" "20020116549" "20030105888" "20030131139" "20040019596" "20040019897" "20040025060"). pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/26 14:29
S3	0	((remov$3 or reduc$3) with bytecode$1 with JAR)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:19
S4	7	(process$3 with JAR with file with target)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:23
S5	4	(JAR with file) same ((map$4 or replac$3) with target with method$1 with name$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:26
S6	5	(JAR with file) and ((map$4 or replac$3) with target with method$1 with name$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:45

S7	78	((map$4 or replac$3) with target with method$1 with name$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:48
S8	5	((map$4 or replac$3) with application$1 with method$1 with shorter with name$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:49
S9	7	((map$4 or replac$3) with method$1 with shorter with name$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 12:49
S10	324	717/118.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 15:51
S11	556	717/120.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 15:51
S12	540	717/151.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 15:51
S13	153	717/166.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 15:51
S14	1918	707/204.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/04/27 15:51

acm PORTAL
USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+jar +global +constant +pool SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Published before February 2004
Terms used **jar** **global** **constant** **pool**

Found **15** of **151,026**

Sort results by: relevance
Display results: expanded form

Save results to a Binder
Search Tips
□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 15 of 15

Relevance scale □ ▤ ▤ ▤ ■

1 Practical extraction techniques for Java ■

Frank Tip, Peter F. Sweeney, Chris Laffra, Aldo Eisma, David Streeter
November 2002 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 24 Issue 6
Publisher: ACM Press

Full text available: pdf(1.01 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Reducing application size is important for software that is distributed via the internet, in order to keep download times manageable, and in the domain of embedded systems, where applications are often stored in (Read-Only or Flash) memory. This paper explores extraction techniques such as the removal of unreachable methods and redundant fields, inlining of method calls, and transformation of the class hierarchy for reducing application size. We implemented a number of extraction techniques in < ...

Keywords: Application extraction, call graph construction, class hierarchy transformation, packaging, whole-program analysis

2 Compressing Java class files ▤

William Pugh
May 1999 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1999 conference on Programming language design and implementation PLDI '99**, Volume 34 Issue 5
Publisher: ACM Press

Full text available: pdf(1.44 MB) Additional Information: full citation, abstract, references, citings, index terms

Java class files are often distributed as jar files, which are collections of individually compressed class files (and possibility other files). Jar files are typically about 1/2 the size of the original class files due to compression. I have developed a wire-code format for collections of Java class files. This format is typically 1/2 to 1/5 of the size of the corresponding compressed jar file (1/4 to 1/10 the size of the original class files).

3 Java bytecode compression for low-end embedded systems ▤

Lars Ræder Clausen, Ulrik Pagh Schultz, Charles Consel, Gilles Muller
May 2000 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 22 Issue 3